Decision Trees/Machine Learning
Akamai CS Inquiry

June 18-19, 2018

GL, GR, LA

Why Inquiry?

Traditional lecture: passive learning

- No “intuitive” understanding
- Misunderstanding of core concepts

This inquiry is intended to

- Help learners get a more intuitive understanding of creating and using models
- Create an environment for you to ask questions and try out ideas comfortably

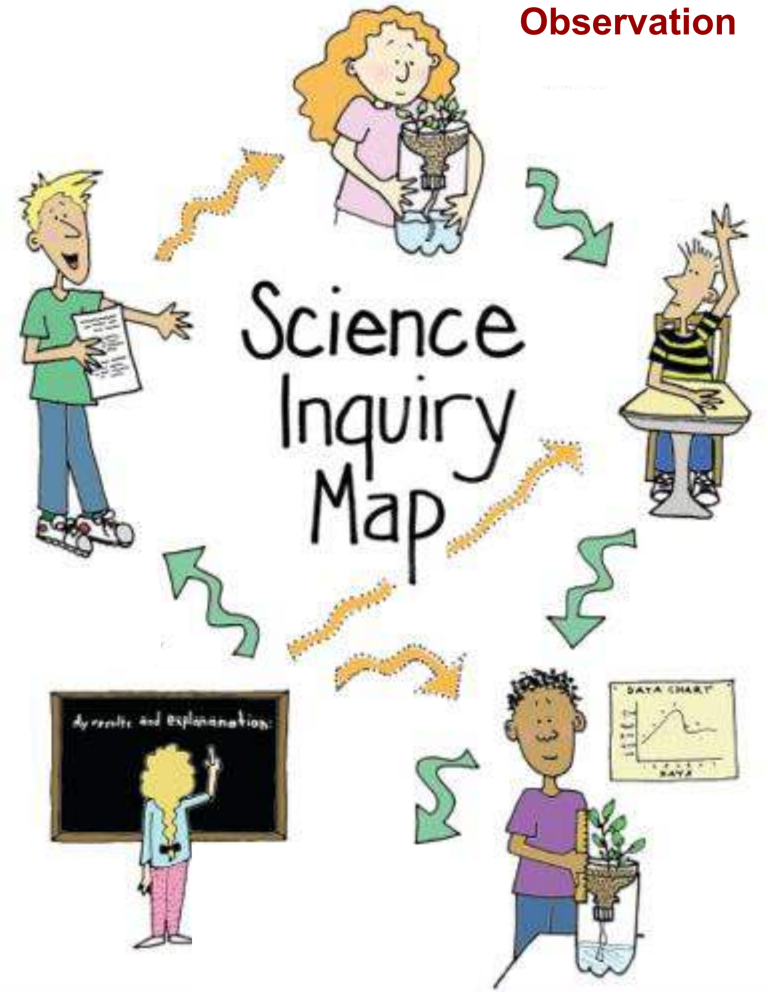
Traditional lecture setting

Active learning environment

Observation

Present

Ask questions



Summarize

Investigate

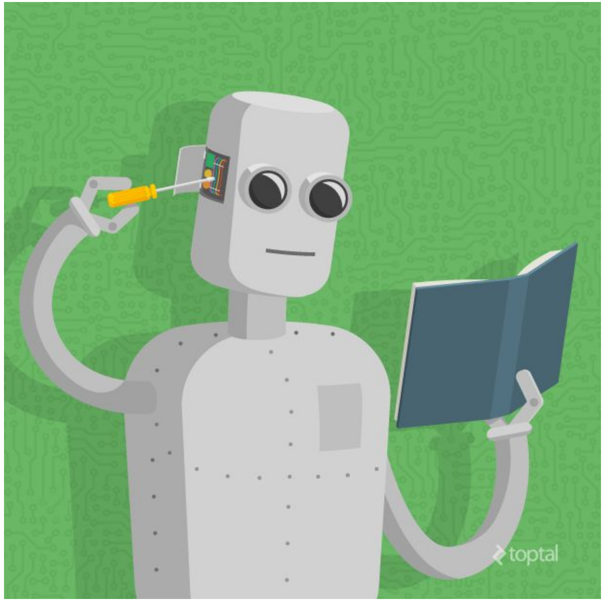
Schedule Overview

1. Introduction
2. Raising questions
3. Focused investigation 1
4. Lunch
5. Focused Investigation 2
6. Preparation for jigsaw presentation
7. Jigsaw with facilitators
8. Synthesis
9. Survey

Estimated time schedule in your handout (electronic).

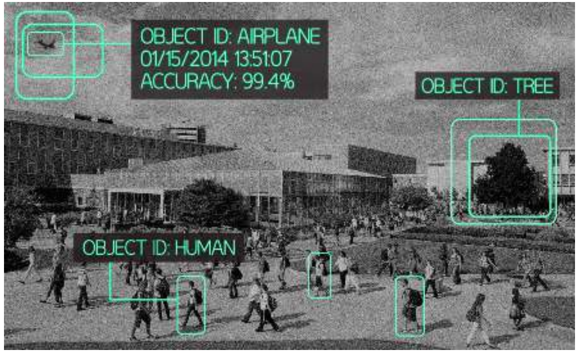
Machine Learning

- Subfield of artificial intelligence
  - Usually part of computer science
- Study of getting computers to learn from and make predictions on data **without being explicitly programmed**
- In general the more data you have the better, but it also depends on the **quality** of your data (garbage in --> garbage out)



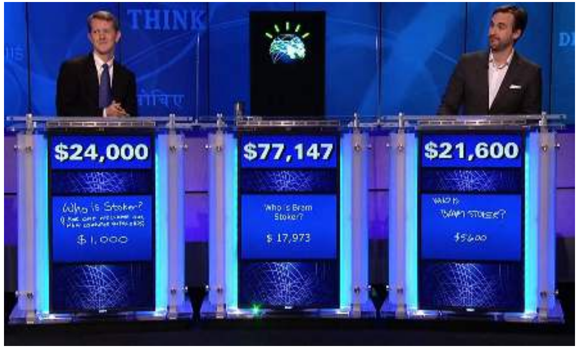
Applications

Robotics/computer vision



Product recommendation

Natural language processing:
IBM Watson on Jeopardy



Video game: bot vs. bot/human

Election results

- Astronomy
- Bioinformatics
- Medical diagnosis
- Stock-market
- Cybersecurity
- Sports analytics
- Law - court cases
- … etc.

FiveThirtyEight

Politics Sports Science & Health Economics Culture

Our Activity

Content Prompt:
Build a **decision tree model** to **accurately predict** a desired attribute of a dataset. Explain or justify how your decision tree solves this predictive problem and maximizes the prediction accuracy.

STEM Practice:
Optimization - Perform tradeoffs between two desirable but often incompatible properties.

For our content: building the “best” model

Source: cambridgeschool.org/blog/growth

- In reality, we are all a **mixture** of fixed and growth mindsets.
- Get closer to growth mindset through **practice**; stay in touch with fixed mindset.
- Cognitive skills can be improved and maintained by **targeted training using increasing demands (challenge).**

Our Activity

- Toy dataset ⇒ real-world dataset
- Provide practical tools and strategies to achieve your goals
- Provide feedback and give suggestions

The goal of this activity is to learn about machine learning through decision trees, and the STEM practice of optimization.

The focus is **not** on programming. But it’s a nice bonus if you learn some programming skills along the way.

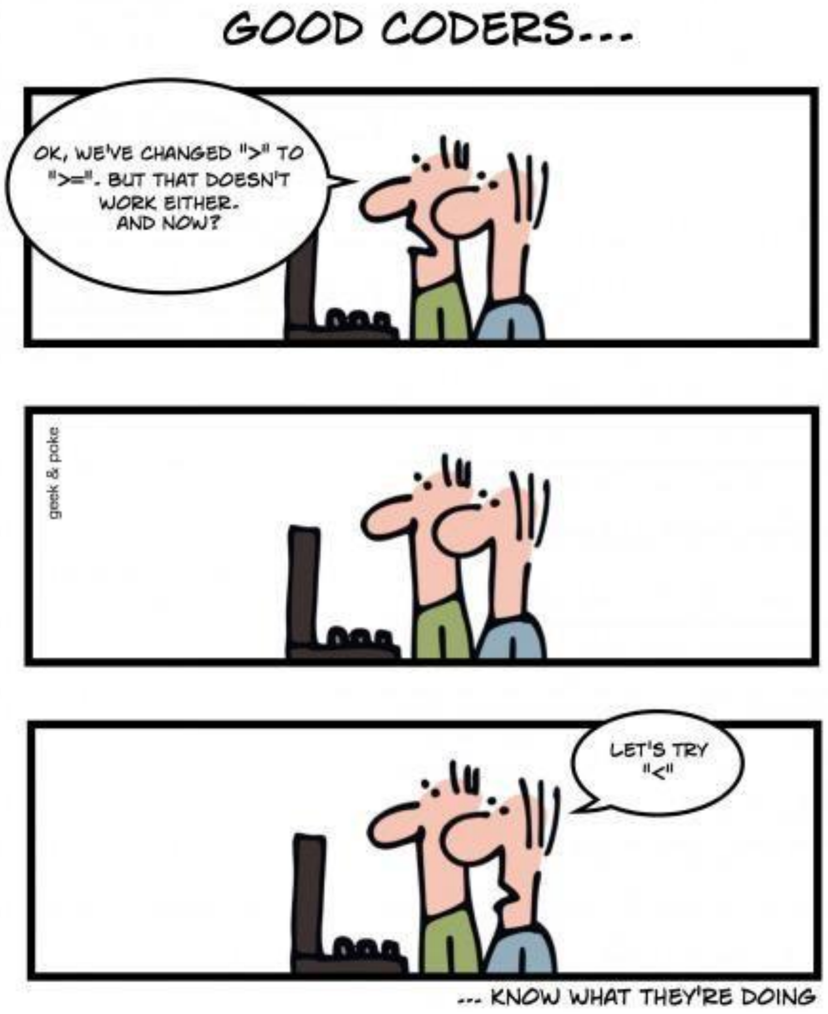
After building and optimizing your model, you are welcome to go above and beyond decision trees. **You can take ownership of your learning!**

Pair Programming

Two programmers at the same station

- **Driver**: writes code
- **Observer**: reviews code as it is typed in

Switch roles frequently



Peer Teaching/Learning

Working together

- Respect differences in background knowledge– don’t just give out answers!

- People navigate inquiry differently– visual vs. vocal learner, etc.
- Be patient with each other

Inquiry Challenges

Investigating is a creative process and you are likely to hit some difficulties as well as breakthroughs. These difficulties are a natural part of the process.

Sometimes you feel you ‘should’ know how something works, but learn that it doesn’t work that way. This is an opportunity to change the way you think about it, and that’s ok.

Getting stuck is a natural part of the process, and can be frustrating. You aren’t doing anything wrong. You’re on the verge of learning something new!

Inquiry Challenges

- It’s okay if the process doesn’t come easily
- If you get stuck, it’s okay to:
  - take a break
  - have a snack
  - get a drink
  - see what other groups are doing

The instructors are your **guides**

- To help you find your way, what to investigate
- But not to spoon-feed you answers

Raising Questions

Decisions in everyday life

Should I get up or sleep in?

Should I stay in school or dropout?

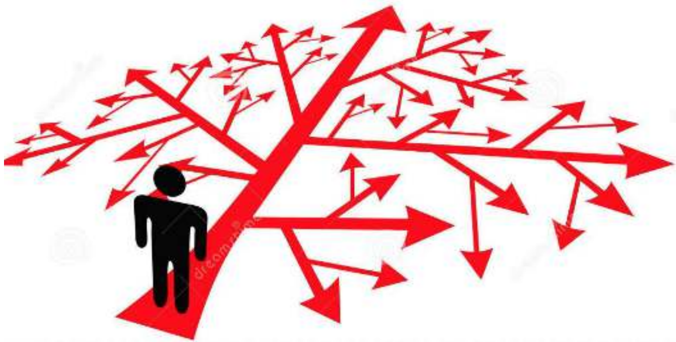
Should I get my coffee at Starbucks or a local coffee shop?

Should I bike or Uber to school?

Should I block him/her on Snapchat?

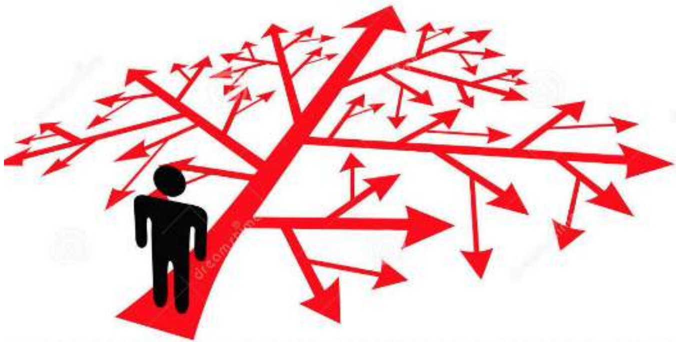
Should I socialize with people or cats?

Should I ask my TA a question or Google it myself?



Decisions in everyday life

Example: **Should I dropout or switch my major?**
What are some **factors**?



Decisions in everyday life

Example: **Should I dropout or switch my major?**

What are some **factors**?

Financial support?

Did Google offer me a six-figure salary?

Have I been here over 5 years?

How many C’s do I have so far?

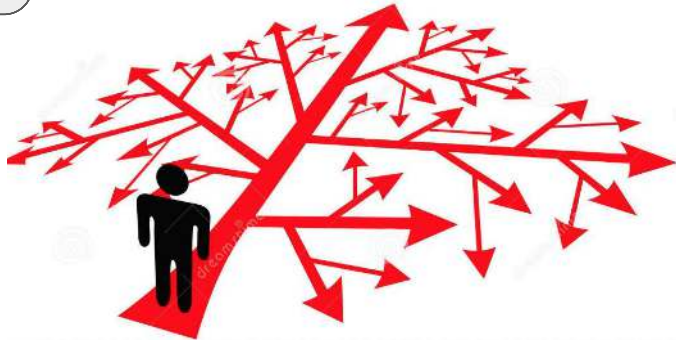
Can I get <u>any</u> job with this major?

Will my parents freak out?

Do I still have friends?

How little do I have to work?

Stress level?



Another example: Data from everyday life

Person	Weather forecast	Company?	Long weekend?	Eateries?	Camping gear?	Outdoorsy?
1	good	no	yes	yes	yes	yes
2	good	yes	Yes	no	yes	yes
3	bad	yes	no	no	yes	no
4	good	yes	yes	no	yes	yes
5	good	yes	no	no	no	yes
6	bad	no	no	yes	no	no

Does it look useful?

What can you do with this data?

Making decisions

Person	Weather forecast	Company?	Long weekend?	Eateries?	Camping gear?	Outdoorsy?
1	good	no	yes	yes	yes	yes
2	good	yes	Yes	no	yes	yes
3	bad	yes	no	no	yes	no
4	good	yes	yes	no	yes	yes
5	good	yes	no	no	no	yes
6	bad	no	no	yes	no	no

Go shopping?

Go fishing?

Eat out?

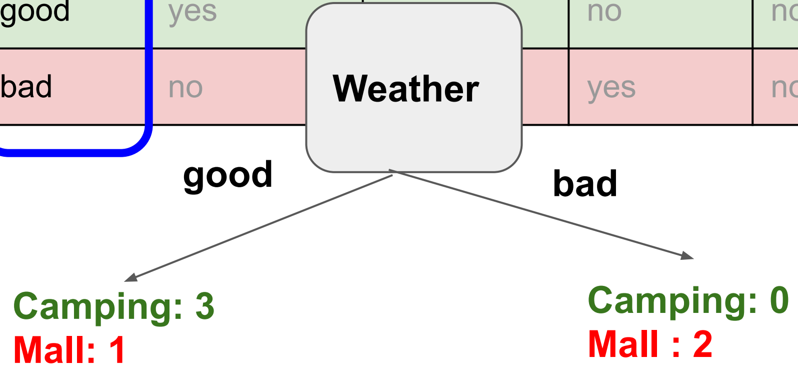
Plan	**Weather forecast**	**Company?**	**Long weekend?**	**Eateries?**	**Camping gear?**	**Outdoorsy?**
mall	good	no	yes	yes	yes	yes
camping	good	yes	Yes	no	yes	yes
mall	bad	yes	no	no	yes	no
camping	good	yes	yes	no	yes	yes
camping	good	yes	no	no	no	yes
mall	bad	no	no	yes	no	no

= Mall

= Camping

Weather

Plan	Weather forecast	Company?	Long weekend?	Eateries?	Camping gear?	Outdoorsy?
mall	good	no	yes	yes	yes	yes
camping	good	yes	Yes	no	yes	yes
mall	bad	yes	no	no	yes	no
camping	good	yes	yes	no	yes	yes
camping	good	yes	[illegible]	no	no	yes
mall	bad	no	[illegible]	yes	no	no



Weather

Plan	Weather forecast	Company?	Long weekend?	Eateries?	Camping gear?	Outdoorsy?
mall	good	no	yes	yes	yes	yes
camping	good	yes	Yes	no	yes	yes
mall	bad	yes	no	no	yes	no
camping	good	yes	yes	no	yes	yes
camping	good	yes		no	no	yes
mall	bad	no		yes	no	

Weather

good

Camping: 3
Mall: 1

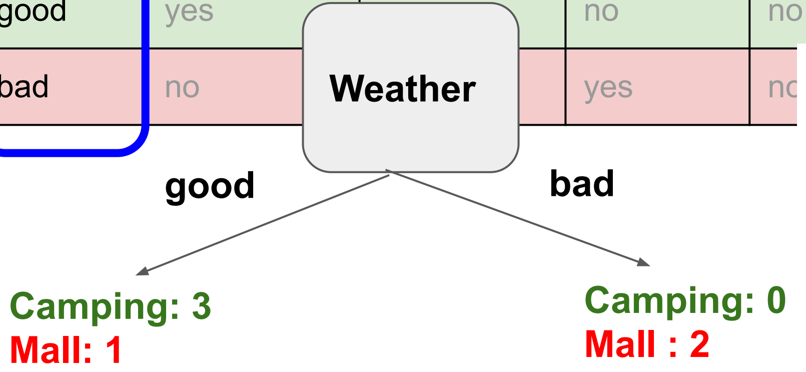
bad

Camping: 0
Mall : 2

If the weather is bad, everyone goes to the mall!

Weather

Plan	Weather forecast	Company?	Long weekend?	Eateries?	Camping gear?	Outdoorsy?
mall	good	no	yes	yes	yes	yes
camping	good	yes	Yes	no	yes	yes
mall	bad	yes	no	no	yes	no
camping	good	yes	yes	no	yes	yes
camping	good	yes	[illegible]	no	no	yes
mall	bad	no	[illegible]	yes	no	[illegible]

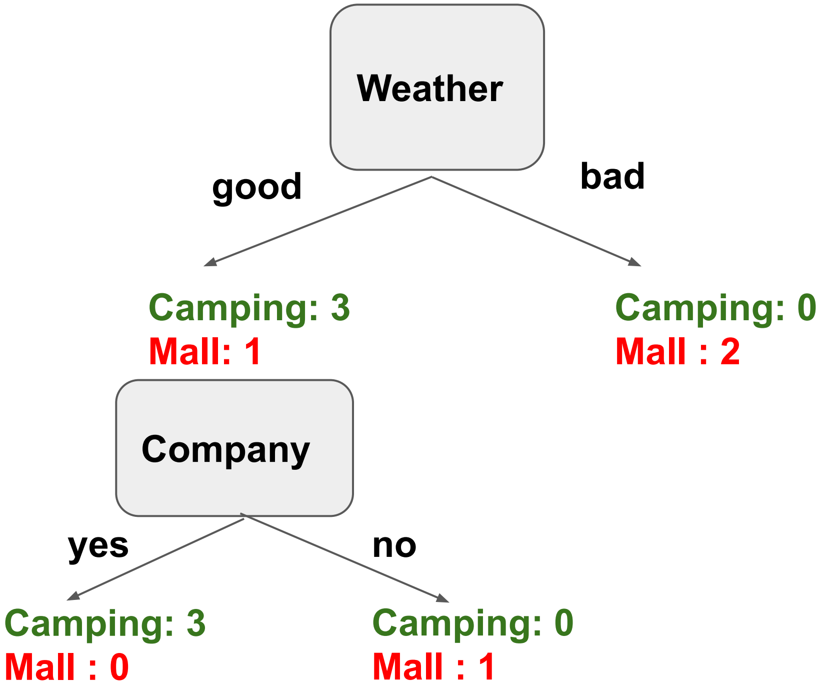


If the weather is bad, everyone goes to the mall!

What happens in good weather?

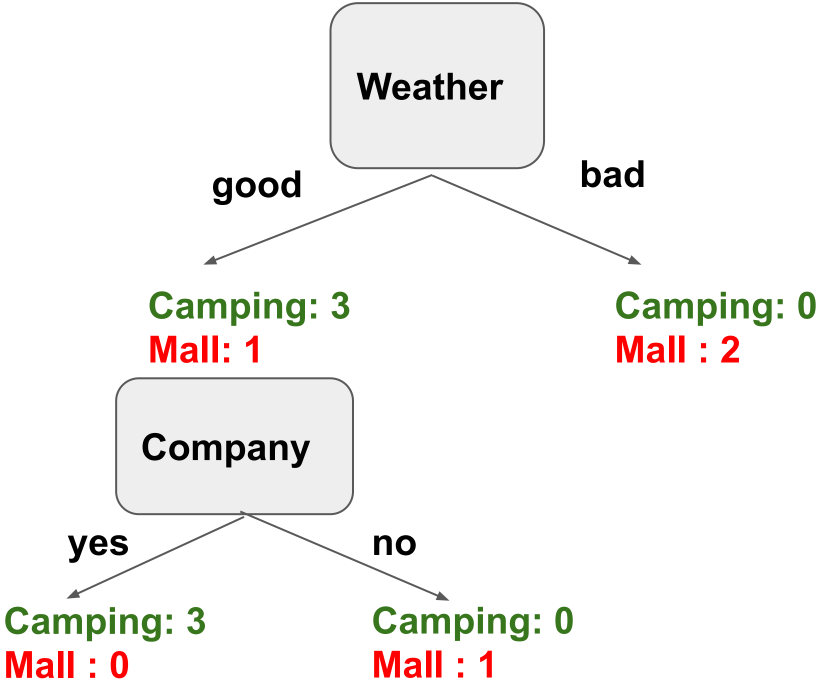
Weather + Company

Plan	Weather forecast	Company?	Long weekend?	Eaterie
mall	good	no	yes	yes
camping	good	yes	Yes	no
mall	bad	yes	no	no
camping	good	yes	yes	no
camping	good	yes	no	no
mall	bad	no	no	yes



Weather + Company

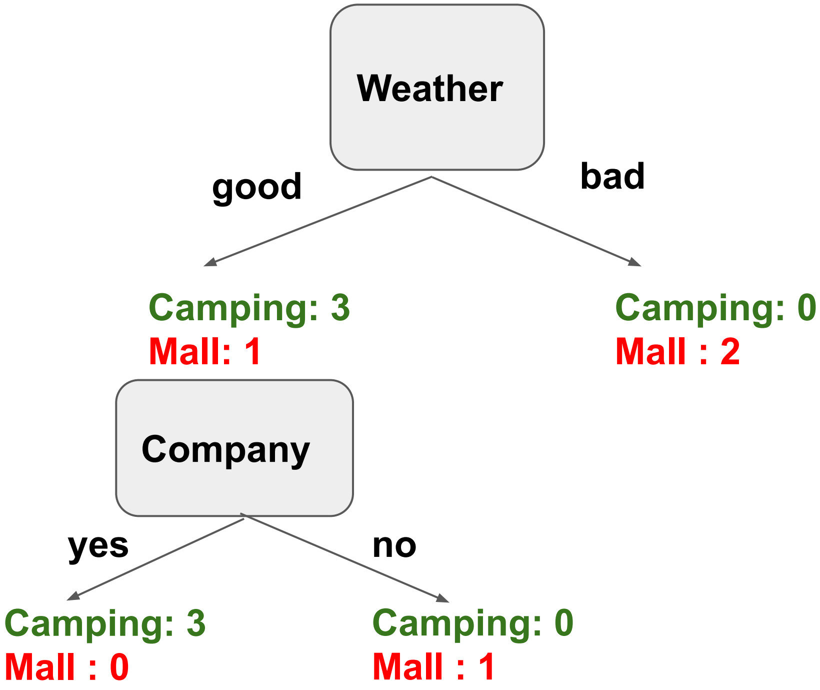
Plan	Weather forecast	Company?	Long weekend?	Eaterie
mall	good	no	yes	yes
camping	good	yes	Yes	no
mall	bad	yes	no	no
camping	good	yes	yes	no
camping	good	yes	no	no
mall	bad	no	no	yes



How would you interpret this model so far?

Weather + Company

Plan	Weather forecast	Company?	Long weekend?	Eaterie
mall	good	no	yes	yes
camping	good	yes	Yes	no
mall	bad	yes	no	no
camping	good	yes	yes	no
camping	good	yes	no	no
mall	bad	no	no	yes



Model interpretation thus far:

- If weather is **bad** --> all **mall**
- If weather is **good** and company is **yes** --> all **camping**
- If weather is **good** and company is **no** --> all **mall**

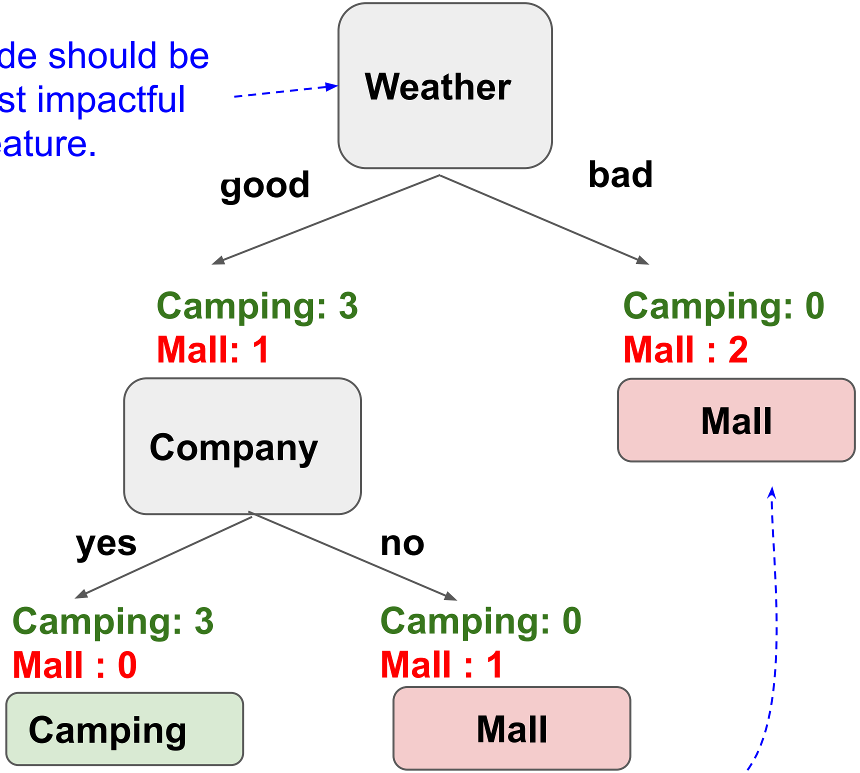
Should we continue to grow the tree?

Weather + Company

Plan	Weather forecast	Company?	Long weekend?
mall	good	no	yes
camping	good	yes	Yes
mall	bad	yes	no
camping	good	yes	yes
camping	good	yes	no
mall	bad	no	no

Suppose that we are done. Then the final model may look like:

Root node should be the most impactful feature.



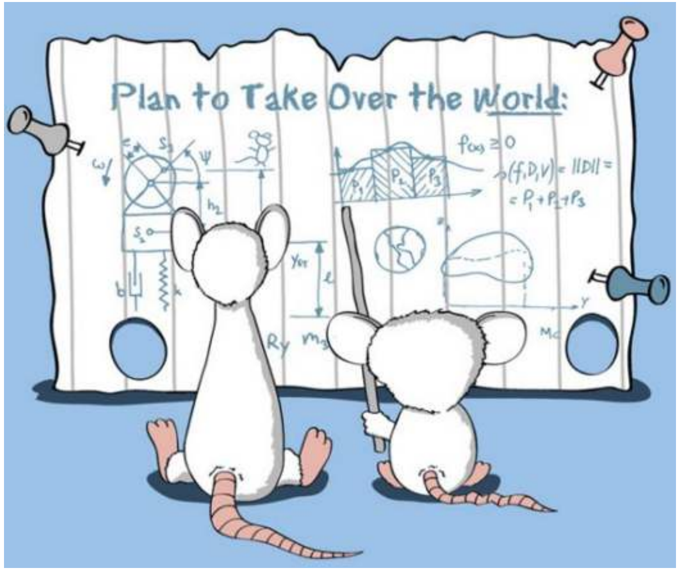
Terminal (**leaf**) nodes are final decisions.

Try on your own! Team formation!

- 2 per team; one team has 3 people.

- At least one person in the group should have programming experience (for afternoon investigation).

- Each team has a **primary facilitator**, but feel free to talk to other facilitators and teams.



Recall our content and practice goal

Content Prompt:
Build a **decision tree** model to **accurately predict** a desired attribute of a dataset. Explain or justify how your decision tree solves this predictive problem and **maximizes** the prediction accuracy.

STEM Practice:
Optimization - Perform tradeoffs between two desirable but often incompatible properties.
(For our content: building the “best” model)

Self-check: content and practice rubrics

Content rubric: decision trees	**Things to notice**
A decision tree predicts a desired attribute of a dataset.	- What is the input and the output of the model? - What are you trying to predict?

The process of choosing the “best” feature to split the tree on at each level.	- How did you decide which feature to use for the root node?
	    - Intuition? Random?
	    - How many features did you pick?
	    - Did you try them all or only a subset?
	- How do you grow the rest of the tree?
	- How good is the model once you decided on a feature for a node?
	- When do you stop growing the tree?

A metric, accuracy, is used to determine model goodness.	- How to calculate it? - Can you calculate it for your model?
Tradeoffs between accuracy and complexity of the model.	- Plot the two properties and observe the relationship. - Can you pick a best model based on trade-off plot? - Why is it the best model?

STEM practice rubric: optimization	**Things to notice**
Describe and use a metric to determine model goodness.	- Describe the metric - what does it mean, how to calculate. - Apply to the model and calculate the results.
Identify and justify important features in the model.	- How to decide whether a feature is important? (Do this at least for the root node for our content)
Perform trade-offs between two desirable but incompatible property of the model to optimize for best model.	- What are the two properties used for trade-off in this case? - What is the relationship between them? - How to optimize for best model using this trade-off?

Electronic Handout

Investigation 1: Build a decision tree model by hand.

Data exploration: (by hand, using Excel, or other spreadsheet-like software)

1. Download **toydata.csv** file from shared Google drive.
2. Open in Excel (your own or through Google drive).
3. Sort and filter as needed. (The “Filter” option in Excel is very useful.)

Build a decision tree model to predict “plan” for the weekend
(i.e. whether you go camping or to the mall)

Think about:

- What is the most predictive feature (for the root node)?
- What is the next most predictive feature (if you decide to grow the tree)?
- When do you stop growing the tree?
- Don’t worry too much about accuracy right now, just try building something! You will get to improve the model later!

Coming up - measuring model goodness

Thinking Tool: Model Performance

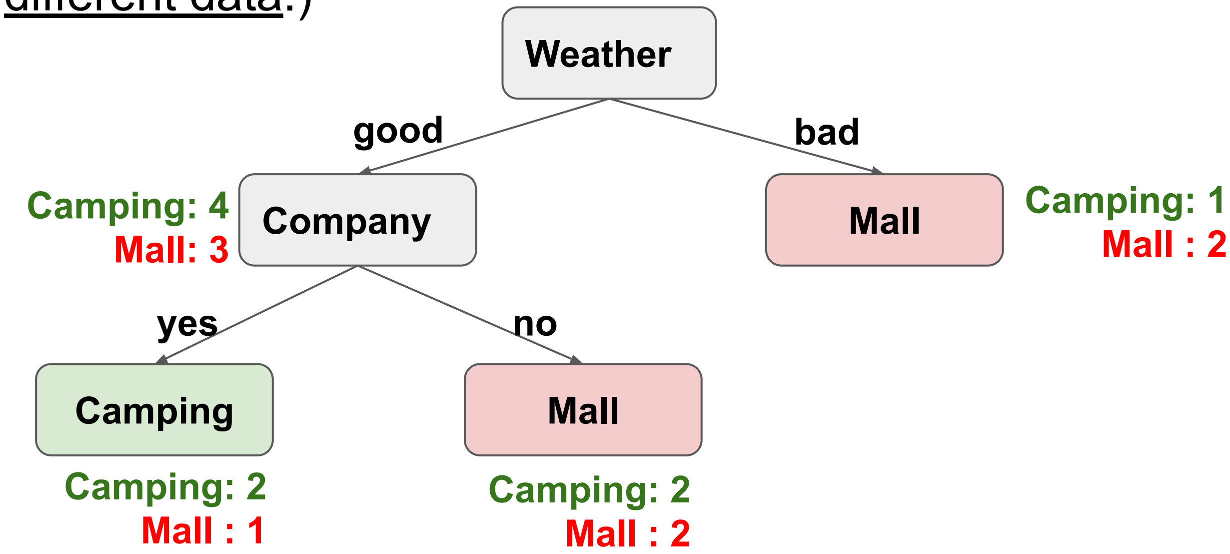
(Same features as before, but different data.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



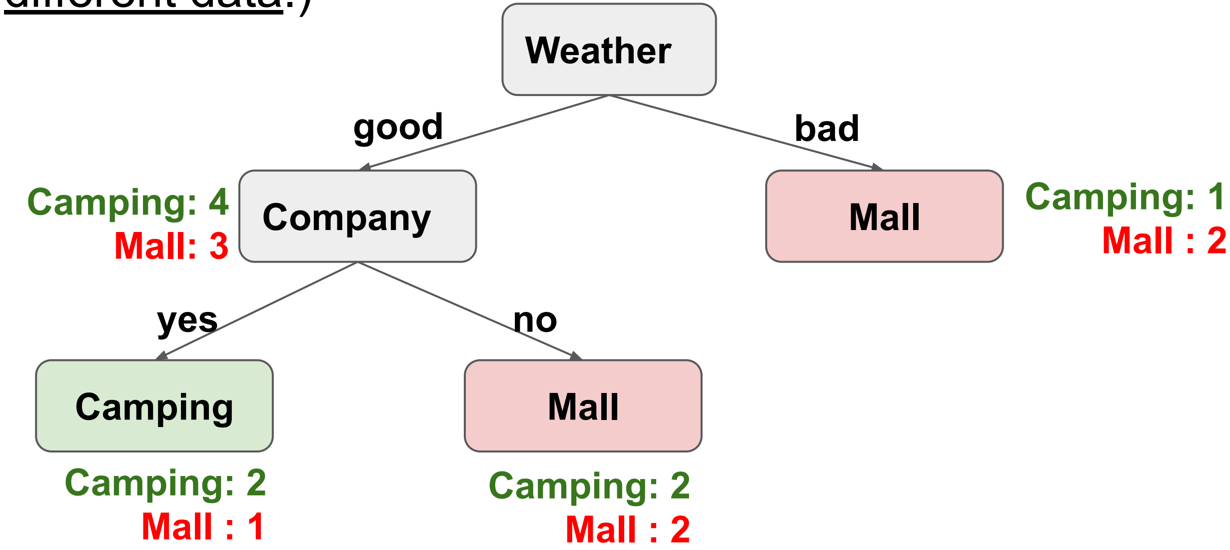
(Same features as before, but <u>different data</u>.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



Overall accuracy

= # correctly predicted samples / total samples

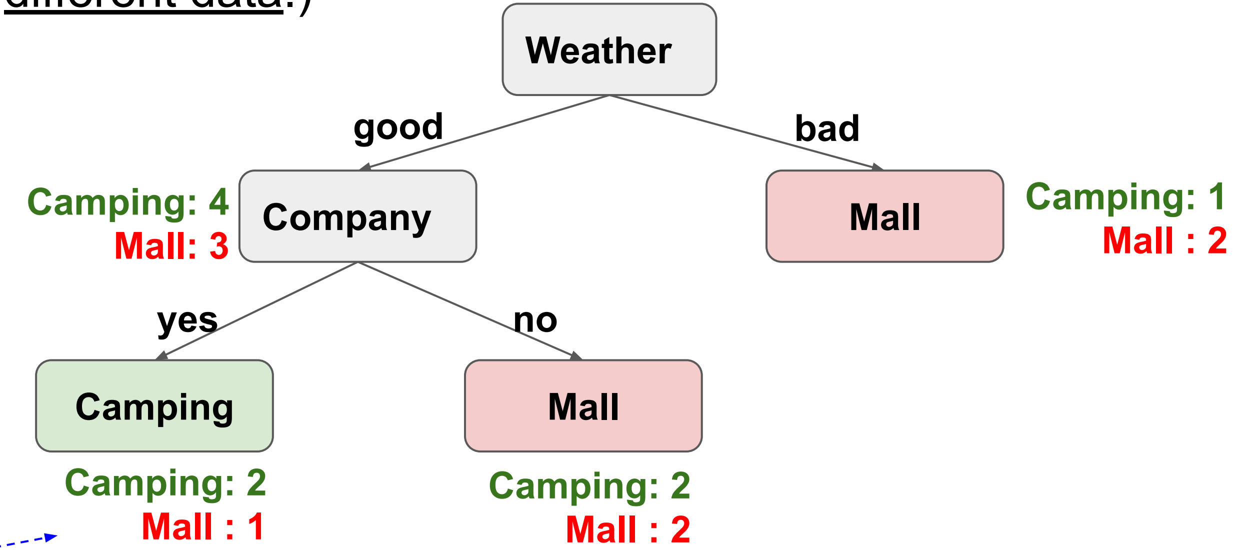
(Same features as before, but different data.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



Predicting **camping**:
correct: 2
incorrect = 1

Overall accuracy
= # correctly predicted samples / total samples

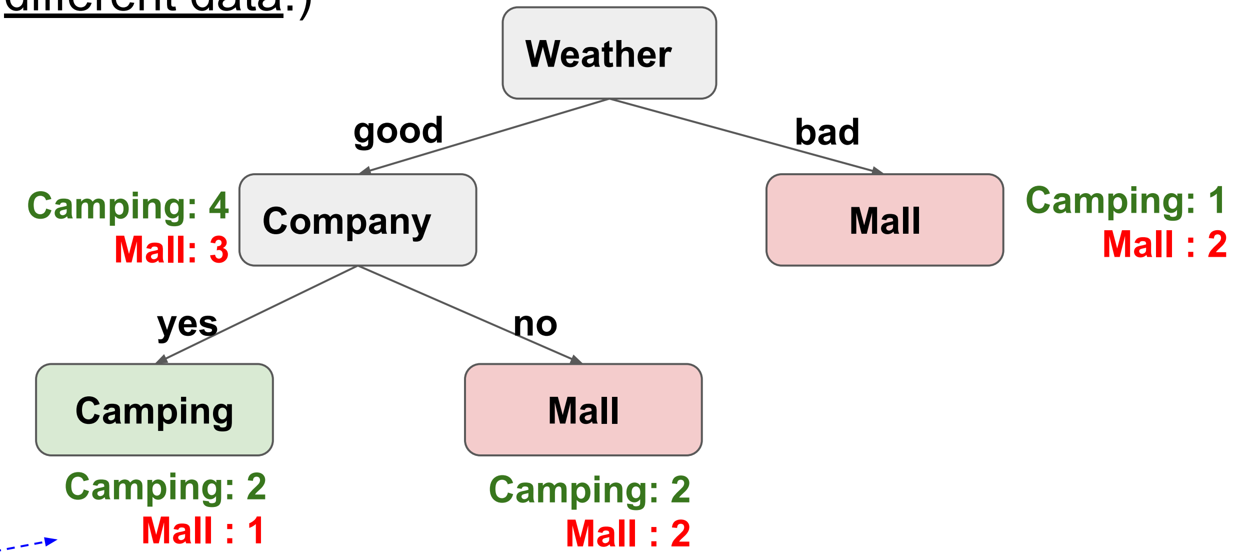
(Same features as before, but different data.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



Predicting **camping**:
\# correct: 2
\# incorrect = 1

Predicting **mall:**
\# correct: 2
\# incorrect = 2

Overall accuracy

= # correctly predicted samples / total samples

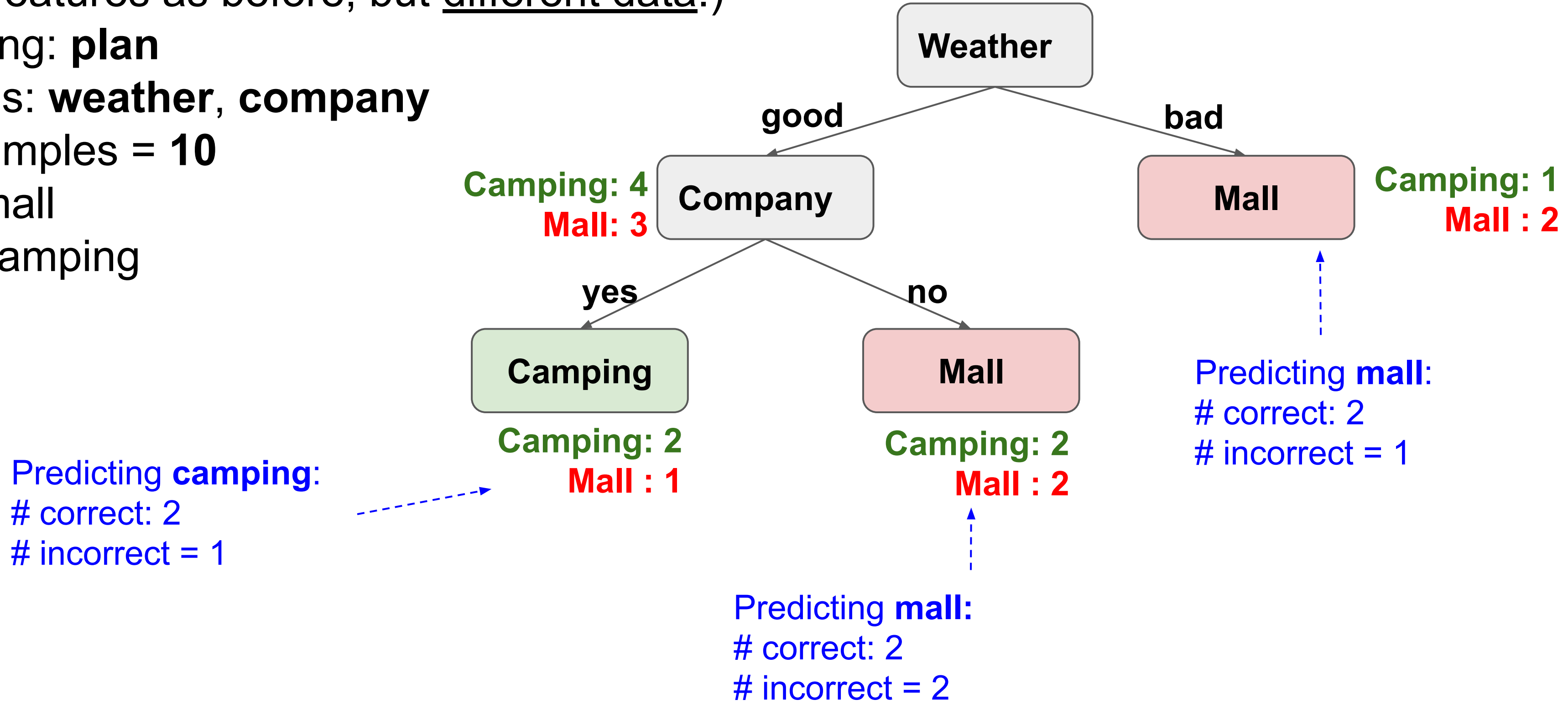
(Same features as before, but <u>different data</u>.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



Overall accuracy

= # correctly predicted samples / total samples

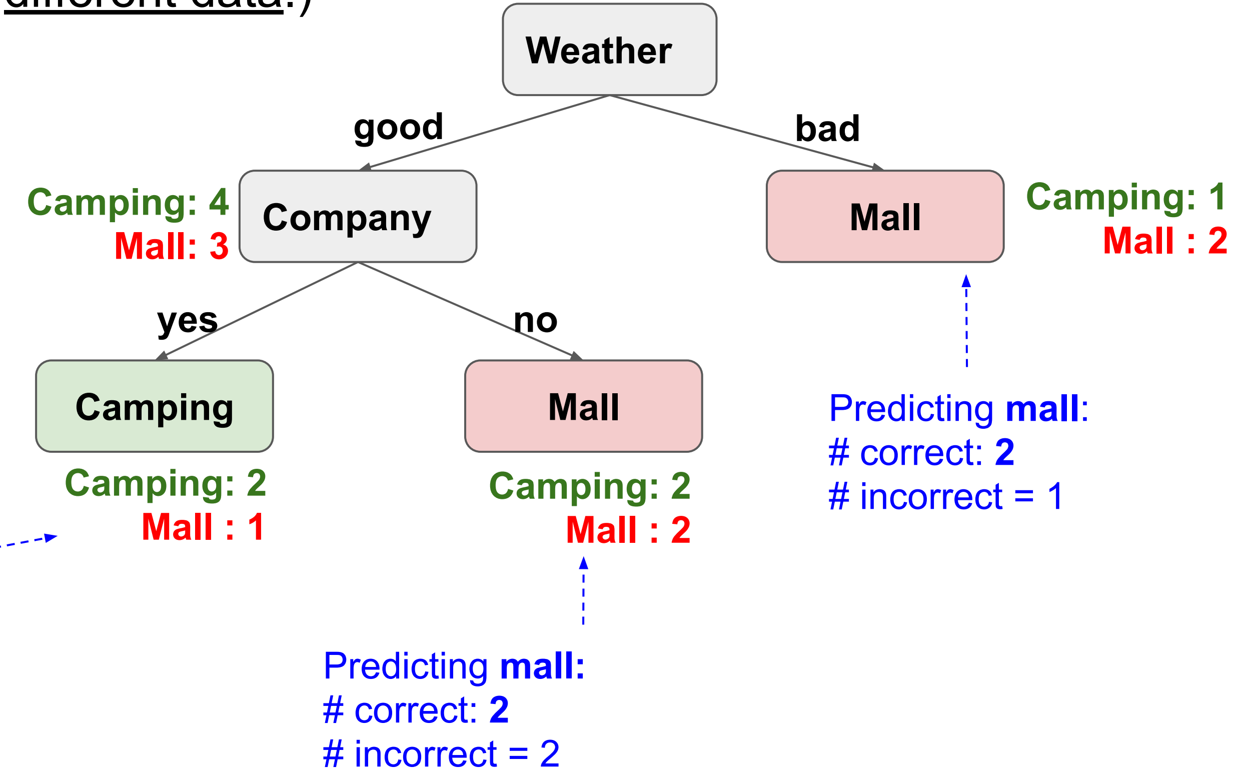
(Same features as before, but <u>different data</u>.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



Predicting **camping**:
correct: **2**
incorrect = 1

Overall accuracy

= # correctly predicted samples / total samples

= (2 + 2 + 2) / 10 = 0.6 = **60%**

Electronic Handout

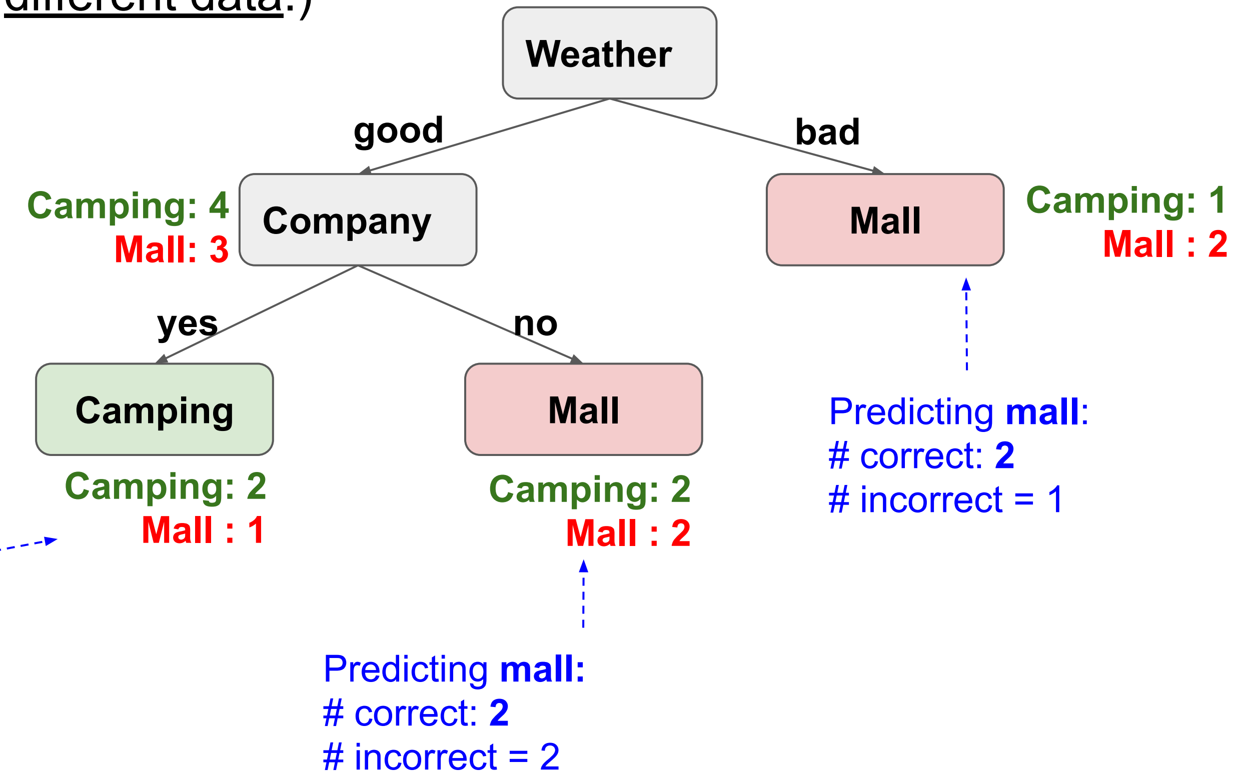
(Same features as before, but different data.)

Predicting: **plan**

Features: **weather**, **company**

Total samples = **10**

- 5 mall
- 5 camping



Predicting **camping**:
correct: **2**
incorrect = 1

Overall accuracy

= # correctly predicted samples / total samples

= (2 + 2 + 2) / 10 = 0.6 = **60%**

Acceptable? Can we improve the model to get a better accuracy?

Low accuracy? No problem! Try some of the following and iterate!

- Add more features.
- Use different features.
- Use a different feature for the root node.
- No need to improve --> what we have is good enough!
- Is your model's accuracy get better or worse as you grow the tree?

Recall our **content prompt**:

- Build a decision tree to accurately predict a desired attribute of a dataset. Explain or justify how your decision tree solves this predictive problem and maximizes the prediction accuracy.

And STEM practice: **optimization**.

Introduction to Investigation 2

Real-world data: **predict house selling price**

18396 houses (rows)

21 features (columns)

Note:

- Easier to predict above/below $ than real values
- For simplicity, we **binarize** the feature using different thresholds
- Your buyers only care about properties that are less than X dollars.
- Choose dataset filename based on the dollar amount you want to split on

	Suburb	Address	Rooms	Type	Price	Method	SellerG
1	Abbotsford	85 Turner St	2	h	1480000	S	Biggin
2	Abbotsford	25 Bloomburg St	2	h	1035000	S	Biggin
4	Abbotsford	5 Charles St	3	h	1465000	SP	Biggin
5	Abbotsford	40 Federation La	3	h	850000	PI	Biggin
6	Abbotsford	55a Park St	4	h	1600000	VB	Nelson
10	Abbotsford	129 Charles St	2	h	941000	S	Jellis
11	Abbotsford	124 Yarra St	3	h	1876000	S	Nelson
14	Abbotsford	98 Charles St	2	h	1636000	S	Nelson
15	Abbotsford	217 Langridge St	3	h	1000000	S	Jellis
16	Abbotsford	18a Mollison St	2	t	745000	S	Jellis
17	Abbotsford	6/241 Nicholson St	1	u	300000	S	Biggin

Pick a binary price you'd like to work on:

- $500K
- $600K
- $700K
- $800K
- $1 million
- $1.2 million
- $1.5 million

PriceBinary1.0M
>=1.0M
>=1.0M
>=1.0M
<1.0M
>=1.0M
<1.0M
>=1.0M
>=1.0M
>=1.0M

Introduction to Investigation 2

Technical spec:

- **Python** - interpreted language (vs. Java/C# are compiled languages)
- **Jupyter** notebook (http://jupyter.org/) - interactive programming framework
- **scikit-learn** library (http://scikit-learn.org/) - a large, open-source Python library containing many ML algorithms.

All can be accessed through **Google's Colaboratory** framework (http://colab.research.google.com)

- A Jupyter notebook environment created to help spread ML education/research. Requires no setup and runs entirely in the cloud.
- Allows collaborations!

Introduction to Investigation 2

All team members:

1. Download the "**code and data for students**" folder we emailed to you. You'll need the data files in this folder on your local drive so you can upload it when you run the notebook.

One team member performs this:

2. Go to your Google Drive.
3. Create a folder for this activity and **share this folder with your teammates**.
4. Upload the **tutorial-partial.ipynb** file from "**code and data for students**" to **your** Google drive folder (which is shared with your teammates).

All team members: Open tutorial-partial.ipynb in Colaboratory

Same notebook, different running session!

Each person on their laptop open up the notebook and start running their own session.

Note: each team member must upload the data to their colaboratory session

To run a cell: **SHIFT + ENTER**

See if you can upload the dataset and run a few cells before going to lunch.

Investigation Time 2: Build decision tree model using Python scikit-learn library with real-world data

Follow prompts in the notebook. Facilitators will be around.

Content Prompt:

Build a **decision tree** model to **accurately predict** a desired attribute of a dataset. Explain or justify how your decision tree solves this predictive problem and **maximizes** the prediction accuracy.

- *Present your final model accuracy and supporting artifacts*
  - *Visualization of the final decision tree model*
  - *Trade-off plot that supports your chosen model*

STEM Practice: Optimization

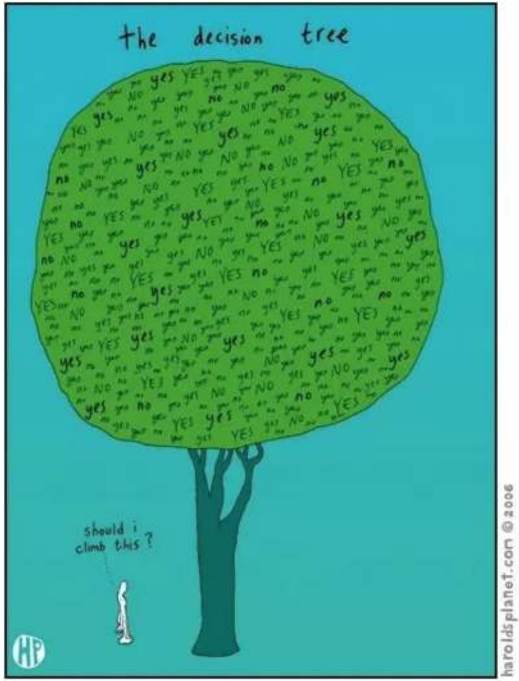
Take turns coding!
(Highly encouraged)

Investigation Time 3: Prediction on new data

1. You will be given a few new samples.
2. Use your best model and make predictions on the new samples.
3. **Does the result make sense? Why or why not?**

This is a **qualitative** evaluation. The idea is to not blindly use results as-is and make sure that we (humans) have a final say in the decision.

- Use your knowledge of the dataset
- Use your prior experience on the subject
- You are welcome to Google any information you like



Jigsaw Preparation

Content Prompt:
Build a decision tree to accurately predict a desired attribute of a dataset. Explain or justify how your decision tree solves this predictive problem and maximizes the prediction accuracy.

- Present your final model accuracy and supporting artifacts
  - Visualization of the final decision tree model
  - Trade-off plot that supports your chosen model
  - *Present your new prediction on the new dataset and summarize your thoughts.*

Jigsaw Presentation

3 groups for 3 facilitators.

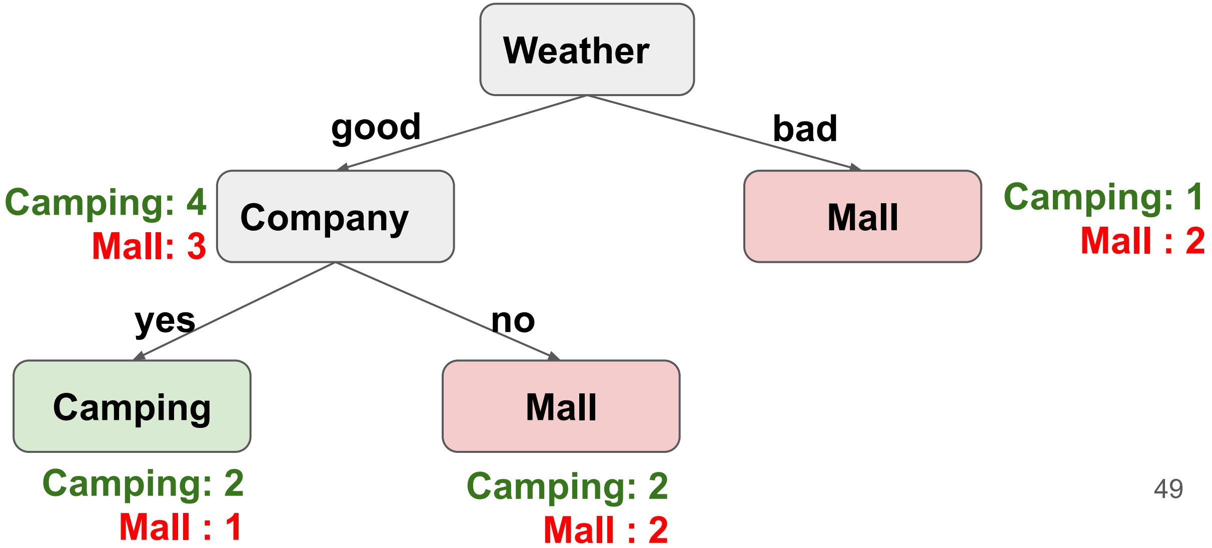
Each group should split up.

Go to a group that’s not your main facilitator.

Synthesis/Wrap-up

What we did... Part 1: Create model by hand

Part 1: Create model by hand
  a. Data exploration by hand or using Excel (or other software)
  b. Pick various features when creating model
  c. Calculate overall accuracy (# predicted correctly / total samples)

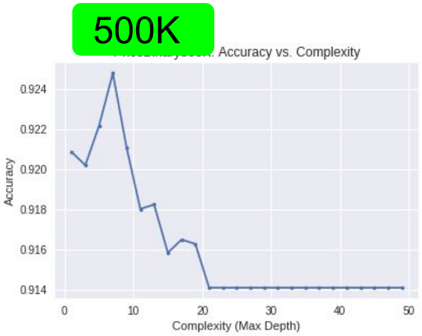


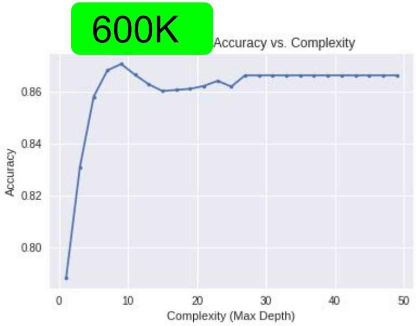
What we did... Part 2: Create model using Python's scikit-learn Decision Tree algorithm

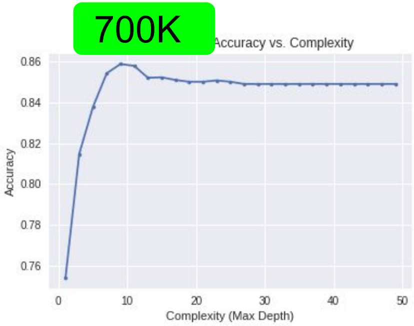
- Separate data into **train/test set** to limit bias
    - Train on "train set"
    - Test on "test set" (unseen)
- Plot **trade-off** between model **accuracy** vs. **complexity** (tree depth)
    - Model complexity increases --> **train** accuracy **increases**
    - Model complexity increases --> **test** accuracy **decreases**
-
- Pick the best model: maximum accuracy while retaining minimum model complexity

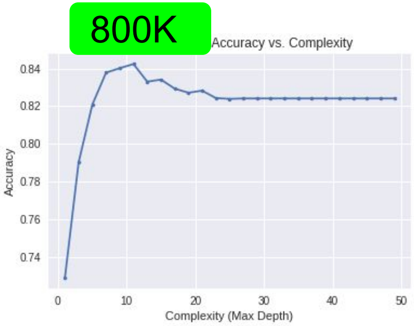
What we did... Part 2: Create model using Python’s scikit-learn Decision Tree algorithm

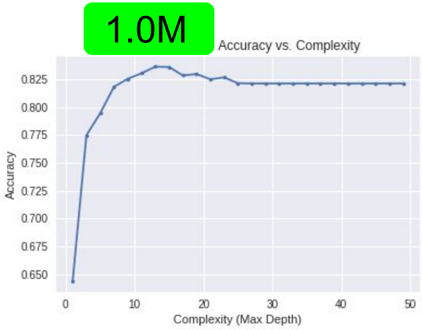
Accuracy vs. model complexity (max tree depth) for different binary targets
['Rooms', 'Bathroom', 'Landsize', 'BuildingArea', 'YearBuilt', 'Lattitude', 'Longtitude']

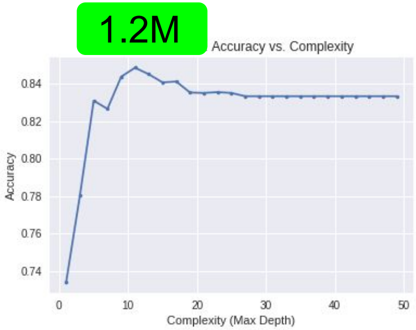


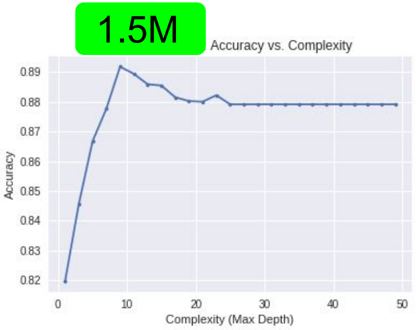












What we did... Part 3: prediction on new data

- Qualitative evaluation of the results
    - Does it make sense? Why or why not?
    - If your model had low prediction accuracy, should you trust the results?
    - If you the model had high prediction accuracy, should you still question the results?

Improving Models / Other ML Algorithms

- Try tuning **other parameters** besides tree depth (e.g., number of nodes)
- Try **other decision-tree-based** of algorithms (Random Forest, XGboost, etc.)
- Try **other non-decision-tree-based** algorithms (SVM, Naive Bayes, etc.)

Same structure of code. Just call different functions for other ML algorithms

Google your questions generously

- All professionals do
- StackOverflow is your best friend

Can you apply decision tree learning (or machine learning in general) to your own project/work?

STEM Practice Reflection

Optimization

1. Describe and use a metric to determine model goodness. Accuracy.
2. Identify and justify important features in the model. In the toy problem, you tried many features and picked the one that gave the best accuracy. Your “gut” feeling may be different from what the data shows (evidence).
3. Perform trade-offs between two desirable but incompatible properties of the model to optimize for best model. Model accuracy vs. complexity (max tree depth).

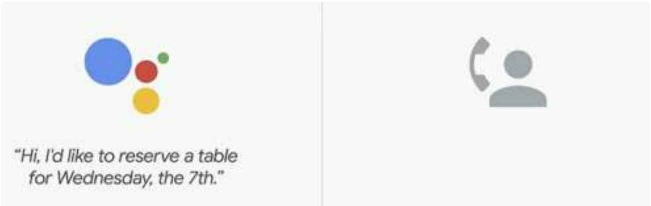
Many ML resources out there!

- Other ML libraries: WEKA (Java), R packages, MATLAB libraries, etc.
- Online courses: Coursera, Udacity, various universities' courses, etc.
- Kaggle tutorials and competitions
- AI podcasts: TWiNL, Data Skeptics, Nvidia AI podcast (deep learning)
- Machine Learning News Google group

ML state of the art examples

AlphaGo (subsequently, AlphaGo Zero, AlphaZero) - first computer Go program to beat a human professional player (Lee Sedol, 9-dan) [March 2016]

Google Duplex (conversation agent) - make calls on behalf of users to schedule appointments or make reservations.



Style Transfer (art)
https://ai.googleblog.com/2016/10/supercharging

Tesla Autopilot (self-driving cars)

Speech generator that can mimic anyone's voice (e.g., startup Lyrebird). Link to listen.

Developing Inclusive ML Algorithms

Example: We want to build a system to classify whether a movie review is positive or negative. We tried 5 different models and found that model C performed the best at this task. We also noticed that model C is most likely to assign a more positive sentiment to the sentence “The main character is a man” than to the sentence “The main character is a woman.”

https://developers.googleblog.com/2018/04/text-embedding-models-contain-bias.html
https://developers.google.com/machine-learning/fairness-overview/

Developing Inclusive ML Algorithms

Example: We want to build a system to classify whether a movie review is positive or negative. We tried 5 different models and found that model C performed the best at this task. We also noticed that model C is most likely to assign a more positive sentiment to the sentence "The main character is a man" than to the sentence "The main character is a woman."

Human data encodes human biases by default.

- Being aware of this is a good start, and the conversation around how to handle it is ongoing.
- To better understand the potential issues that an ML model might create, both **model creators** and **practitioners** who use these models should examine the undesirable biases that models may contain.

https://developers.googleblog.com/2018/04/text-embedding-models-contain-bias.html
https://developers.google.com/machine-learning/fairness-overview/

As AI/Machine learning becomes more mainstream, keep in mind the **social impact** of your application!

Think about possible **bias** in data.

How will the trained model be **useful**
- to showcase current state of things (i.e., recognizing the bias in data).

How will the trained model be **harmful**
- to predict/decide on new information (i.e., propagating the bias).

https://xkcd.com/1838/


"Tell us what you think!"